

# Отчет по лабораторной работе №2

Валеев Нурсан М4139

## Описание алгоритма

Была использована следующая цепочка алгоритмов:

**BWT** -> **MTF** (стопка книг) -> **Адаптивное арифметическое** кодирование с использованием предыдущих **n** бит

Так как  $H(X|X)$  и  $H(XX|X)$  значительно меньше для каждого файла из условия чем  $H(X)$ , было использовано кодирование с контекстом. Но так как оно адаптивное (без записи статистики в файл), его выигрыш может сильно отличаться от контекста к контексту (или даже хуже сжимать). Поэтому был сделан перебор по контекстам 1 -> 16 бит и выбран лучший (он для каждого файла пишется внутри файла и декодер читая его, знает с каким контекстом был закодирован исходный файл).

## Результаты работы

В таблице приведены размеры сжатых файлов

Файл	jpeg30	jpeg80
airplane	18799	43963
arctichare	11583	26293
baboon	35925	88239
cat	35753	81798
fruits	17481	45232
frymire	195735	434421
girl	23499	59544
lena	17418	43754

Файл	jpeg30	jpeg80
monarch	28138	63939
peppers	18622	47792
pool	6264	13825
sails	45094	105567
serrano	57514	136994
tulips	37769	85611
watch	44661	99701

Суммарный размер сжатых файлов (jpeg30) - 594255 байт. (> 2%)

Суммарный размер сжатых файлов (jpeg80) - 1376674 байт. (< 2%)

## Приложение

---

Программа **coder.exe**

coder.exe input output

Программа **decoder.exe**

coder.exe input output

Исходные файлы в виде zip архива **source.zip**